Query-focused Multi-Document Summarization: Combining a Topic Model with Graph-based Semi-supervised Learning

Yanran Li and **Sujian Li***
Key Laboratory of Computational Linguistics,
Peking University, MOE, China
{liyanran,lisujian}@pku.edu.cn

Abstract

Graph-based learning algorithms have been shown to be an effective approach for query-focused multi-document summarization (MDS). In this paper, we extend the standard graph ranking algorithm by proposing a two-layer (i.e. sentence layer and topic layer) graph-based semi-supervised learning approach based on topic modeling techniques. Experimental results on TAC datasets show that by considering topic information, we can effectively improve the summary performance.

1 Introduction

Query-focused multi-document summarization (MDS) can facilitate users to grasp the main idea of the documents according to the users' concern. In query-focused summarization, one query is firstly proposed at the beginning of the documents. Then according to the given query and its influence on sentences, a ranking score is assigned to each of the sentences and higher ranked sentences are picked into a summary.

Among existing approaches, graph-based semi-supervised learning algorithms have been shown to be an effective way to impose a query's influence on sentences (Zhou et al, 2003; Zhou et al, 2004; Wan et al, 2007). Specifically, a weighted network is constructed where each sentence is modeled as a node and relationships between sentences are modeled as directed or undirected edges. With the assumption that a query is the most important node, initially, a positive score is assigned to the query and zero to the remaining nodes. All nodes then spread their ranking scores to their nearby neighbors via the weighted network. This spreading process is repeated until a global stable state is achieved, and all nodes obtain their final ranking scores.

The primary disadvantage of existing learning method is that sentences are ranked without considering topic level information. As we know, a collection of related documents usually covers a few different topics. For example, the specific event "Quebec independence" may involve the topics such as "leader in independence movement", "referendum", "related efforts in independence movement" and so on. It is important to discover the latent topics when summarizing a document collection, because sentences in an important topic would be more important than those talking about trivial topics (Hardy et al, 2002; Harabagiu and Lacatusu, 2005; Otterbacher et al, 2005; Wan and Yang, 2008).

The topic models (Blei et al, 2003) offer a good opportunity for the topic-level information modeling by offering clear and rigorous probabilistic interpretations over other existing clustering techniques. So far, LDA has been widely used in summarization task by discovering topics latent in the document collections (Daume and Marcu, 2006; Haghighi and Vanderwende, 2009; Jin et al, 2010; Mason and Charniak, 2011; Delort and Alfonseca, 2012). However, as far as we know, how to combine topic information and semi-supervised learning into a unified framework has seldom been exploited.

In this paper, inspired by the graph-based semi-supervised strategy and topic models, we propose a two-layer (i.e. sentence layer and topic layer) graph-based semi-supervised learning approach for

*correspondence author

query-focused MDS. By using two revised versions of LDA topic model (See Section 2), our approach naturally models the relations between topics and sentences, and further use these relations to construct the two-layer graph. Experiments on the TAC datasets demonstrate that we can improve summarization performance under the framework of two-layer graph-based semi-supervised learning.

The rest of this paper is organized as follows: Section 2 describes our LDA based topic models, W-LDA and S-LDA. Section 3 presents the construction of the two-layer graph and the semi-supervised learning and the experimental results are provided in Section 4. Then, Section 5 describes related work on query-focused multi-document summarization and topic modeling techniques and we conclude this paper in Section 6.

2 Topic Modeling

2.1 Model Description

As discussed in Section 1, a collection of documents often involves different topics related to a specific event. The basic idea of our summarization approach is to discover the latent topics and cluster sentences according to the topics. Inspired by (Chemudugunta et al, 2006) and (Li et al, 2011), we find 4 types of words in the text: (1) Stop words that occur frequently in the text. (2) Background words that describe the general information about an event, such as ”Quebec” and ”independence”. (3) Aspect words talking about topics across the corpus. (4) Document-specific words that are local to a single document and do not appear across different corpus. Similar ideas can also be found in many LDA based summarization techniques (Haghighi and Vanderwende, 2009; Li et al, 2011; Delort and Alfonseca, 2012).

Stop words can easily be filtered out by a standard list of stopwords. We use a background word distribution ϕ_B to model vocabularies commonly used in the document collection. We assume that there are K aspect topics shared across corpus and each topic is associated with a topic-word distribution $\phi_k,\ k \in [1, K]$. For each document m, there is a document-specific word distribution $\phi_m,\ m \in [K+1, K+M]$. Each word w is modeled as a mixture of background topics, document-specific topics or aspect topics. We use a latent parameter y_w to denote whether it is a background word, a document-specific word or an aspect word. y_w is sampled from a multinomial distribution with parameter π.

2.2 W-LDA and S-LDA

We describe two models: a word level model W-LDA and a sentence level S-LDA. Their difference only lies in whether the words within a sentence are generated from the same topic.

W-LDA: Figure 1 and Figure 3 show the graphical model and generation process of W-LDA, which is based on Chemudugunta et al's work (2007). Using the Gibbs sampling technique, in each iteration two latent parameters y_w and z_w are sampled simultaneously as follows:

$$P(y_w = 0) \propto \frac{N_{m0,-w} + \gamma}{N_{m,-w} + 3\gamma} \frac{E_B^w + \lambda}{\sum_{w'} E_B^{w'} + V\lambda} \tag{1}$$

$$P(y_w = 1) \propto \frac{N_{m1,-w} + \gamma}{N_{m,-w} + 3\gamma} \frac{E_m^w + \lambda}{\sum_{w'} E_m^{w'} + V\lambda} \tag{2}$$

$$P(y_w = 2, z_w = k) \propto \frac{N_{m2,-w} + \gamma}{N_{m,-w} + 3\gamma} \times \frac{C_m^k + \alpha}{\sum_{k'} C_m^k + K\alpha} \frac{E_k^w + \lambda}{\sum_{w'} E_k^{w'} + V\lambda} \tag{3}$$

where $N_{m0,-w}, N_{m1,-w}$ and $N_{m2,-w}$ denote the number of words assigned to background, document-specific and aspect topic in current document. $N_{m,-w}$ denotes the total number of words in current document. E_B^w, E_m^w and E_k^w are the number of times that word w appears in background topic, document-specific topic and aspect topic k. C_m^k denotes the number of words assigned to topic k in current document.

With one Gibbs sampling, we can make the following estimation:

$$\phi_k^w = \frac{E_k^w + \lambda}{\sum_{w'} E_k^{w'} + V\lambda} \tag{4}$$

Then, the probability that a sentence s is generated from topic k is computed based on the probability that each of its aspect words is generated from topic k:

$$P(s|z_s = k) = \prod_{w \in s, y_w = 2} \phi_k^w \tag{5}$$

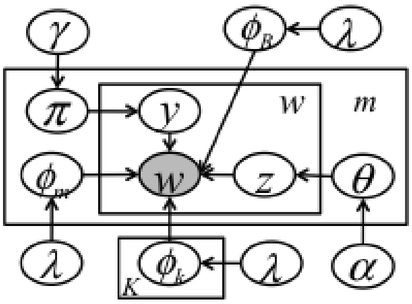


Figure 1: Graphical model for W-LDA

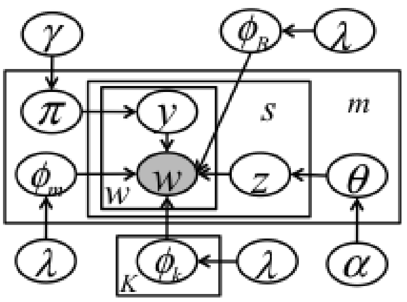


Figure 2: Graphical model for S-LDA

```
1. Draw background distribution $\phi_B \sim Dir(\lambda)$
2. For each document $m$:
      draw doc proportion vector $\theta_m \sim Dir(\alpha)$
      draw doc proportion vector $\pi_m \sim Dir(\gamma)$
      draw doc specific distribution $\phi_m \sim Dir(\lambda)$
3. For each topic $k$:
      draw topic distribution $\phi_k \sim Dir(\lambda)$
4. For each word $w$ in document $m$:
   (a) draw $y_w \sim Multi(\pi_m)$
   (b) if $y_w = 0$: draw $w \sim \phi_B$
       if $y_w = 1$: draw $w \sim \phi_m$
       if $y_w = 2$:
           draw $z_w \sim Multi(\theta_m)$
                $w \sim Multi(\phi_{z_w})$
```

Figure 3: Generation process for W-LDA

```
1. Draw background distribution $\phi_B \sim Dir(\lambda)$
2. For each document $m$:
      draw doc proportion vector $\theta_m \sim Dir(\alpha)$
      draw doc proportion vector $\pi_m \sim Dir(\gamma)$
      draw doc specific distribution $\phi_m \sim Dir(\lambda)$
3. For each topic $k$:
      draw topic distribution $\phi_k \sim Dir(\lambda)$
4. For each sentence $s$ in document $m$:
   4.1 draw $z_s \sim Multi(\theta_m)$
   4.2 for each word in sentence $s$:
       (a) draw $y_w \sim Multi(\pi_m)$
       (b) if $y_w = 0$: draw $w \sim \phi_B$
           if $y_w = 1$: draw $w \sim \phi_m$
           if $y_w = 2$: draw $w \sim Multi(\phi_{z_w})$
```

Figure 4: Generation process of S-LDA

S-LDA: In S-LDA, each sentence is treated as a whole and words within a sentence are generated from the same topic (Gruber et al., 2007). Its graphical model and generated process are shown in Figure 2 and Figure 4. In S-LDA, we firstly sample the topic z_s for each sentence as follows:

$$\begin{aligned} P(z_s = k|z_{-s}, y, w) &\propto \frac{\Gamma(\sum_{w'} E_k^{w'} + V\lambda)}{\Gamma(\sum_{w'} E_k^{w'} + N_s^A + V\lambda)} \\ &\times \prod_{w \in s, y_w = 2} \frac{\Gamma(E_k^w + N_s^w + \lambda)}{\Gamma(E_k^w + \lambda)} \cdot \frac{C_m^k + \alpha}{\sum_{k'} C_m^{k'} + K\alpha} \end{aligned} \tag{6}$$

C_m^k denotes the number of sentences in document m assigned to topic k. N_s^A denotes the number of aspect words in current sentence. Then y_w is sampled.

In our experiments, we set hyperparameters $\alpha = 1,\ \beta = 0.5,\ \lambda = 0.01$. We run 500 burn-in iterations through all documents in the collection to stabilize the distribution of z and y before collecting samples.

3 Graph-based Semi-supervised Learning

As stated before, the consideration of higher level information (i.e. topics) would be helpful for sentence ranking in summarization. In our two-layer graph, the upper layer is composed of topic nodes and the lower layer is composed of sentences nodes, among which there is one node representing the query.

Formally, given a document set D, let $G =< V_s, V_t, E >$ be the two-layer graph, where $V_s = \{s_1, s_2, ..., s_N\}$ denotes the set of all the sentence nodes and s_1 is the query. $V_t = \{z_1, z_2, ..., z_K\}$ corresponds to all the topic nodes. The collection of edges E in the graph consists of the relations within layers and between layers. And the edge weights are measured according to the similarities between nodes, which are computed based on the topic distribution from our two topic model extensions. Specifically, we introduce four edge weight matrices $\hat{W}_{N*K}$, $\bar{W}_{K*N}$, U and P to describe the sentence-to-topic relations, the topic-to-sentence relations, the sentence-to-sentence relations and the topic-to-topic relations respectively.

Firstly, the row-normalized edge weight matrices $\hat{W}_{N*K}$ and $\bar{W}_{K*N}$ denotes the similarity matrix between sentences and topics,

$$\hat{W}_{i,j} = \frac{sim(s_i, z_k)}{\sum_{k'} sim(s_i, z_{k'})} \quad \bar{W}_{i,j} = \frac{sim(s_i, z_k)}{\sum_j sim(s_j, z_k)} \tag{7}$$

where $sim(s_i, z_k) = p(s_i|z_{s_i} = z_k)$ is the probability that the sentence is generated from that topic calculated in Equation (5).

The edge weight matrix U describe the sentence-to-sentence relations. In the same way, the similarity between two sentences is the cosine similarity between their topic distributions, $sim(s_i, s_j) = \frac{1}{C_1} \sum_k p(s_i|z_{s_i} = k) \cdot p(s_j|z_{s_j} = z_k)$, where $C_1 = \sqrt{\sum_k p^2(s_i|z_{s_i} = k)}\sqrt{\sum_k p^2(s_j|z_{s_j} = k)}$ is the normalized factor. Since the row-normalization process will make the sentence-to-sentence relation matrix asymmetric, we adopt the following strategy: let $Sim(s)$ denote the similarity matrix between sentences, where $Sim(s)(i, j) = sim(s_i, s_j)$ and D denotes the diagonal matrix with (i, i)-element equal to the sum of the i^{th} row of $Sim(s)$. Edge weight matrix between sentences U is calculated as follows:

$$U = D^{-\frac{1}{2}} Sim(s) D^{-\frac{1}{2}} \tag{8}$$

Then, the edge weight matrix between topics P is the normalized symmetric matrix of the similairty matrix between two topics. The cosine similarity between two topics is calculated according to word-topic distribution.

$$sim(z_i, z_j) = \frac{1}{C_2} \sum_w p(w|z_i) p(w|z_j) = \frac{1}{C_2} \sum_w \phi_{z_i}^w \phi_{z_j}^w \tag{9}$$

where $C_2 = \sqrt{\sum_w p^2(w|z_i)} \cdot \sqrt{\sum_w p^2(w|z_j)}$ is the normalized factor.

We further transform the task to an optimizing problem based on the assumption that closely related nodes (sentences and topics) tend to have similar scores. So we would give more penalty for the difference between closely related nodes with regard to edge weight matrices $\hat{W}_{N*K}$, $\bar{W}_{K*N}$, U and P. This motivates the following optimization function $\Omega(f, g)$ in Equation (10) similar to the graph harmonic function(Zhu et al, 2003). f denotes the sentence score vector and g denotes the topic score vector. Intuitively, $\Omega(f, g)$ measures the sum of difference between graph nodes; the more they differ, the larger $\Omega(f, g)$ would be.

$$\begin{aligned} \Omega(f, g) = a \sum_{0 \le i,j \le N} U_{i,j} (f_i - f_j)^2 + a \sum_{0 \le i,j \le K} P_{i,j} (g_i - g_j)^2 \\ + (1 - a) \sum_{0 \le i \le N} \sum_{0 \le j \le K} \hat{W}_{ij} (f_i - g_j)^2 \\ + (1 - a) \sum_{0 \le i \le N} \sum_{0 \le j \le K} \bar{W}_{ij} (g_i - f_j)^2 \end{aligned} \tag{10}$$

The score vectors can be achieved by minimizing the function in Equation (10). That is, (f, g)=argmin$_{f,g}\Omega(f, g)$. We can get the following equations (details are shown in Appendix).

$$\begin{aligned} f = aUf + \frac{1}{2}(1 - a)(\hat{W} + \bar{W}^T) g \\ g = aPg + \frac{1}{2}(1 - a)(\hat{W}^T + \bar{W}) f \end{aligned} \tag{11}$$

Equation (11) conforms to our intuition: (1) A sentence would be important if it is heavily connected with many important sentences and a topic would be important if it is closely related to other important topics. (2) A sentence would be important if it is expressing an important topic, and in turn a topic would be important if it is referred by an important sentence. Based on Equation (11), the ranking algorithm is designed in a semi-supervised way, where the score of the labeled query is fixed to the largest score of 1 during each iteration, as shown in Figure 5. Then, our algorithm iteratively calculates the score of topics and sentences until convergence[1].

```
Input: The sentence set $\{s_1, s_2, ..., s_N\}$, topic set $\{z_1, z_2, ..., z_K\}$, edge weight matrix $\hat{W}$, $\bar{W}$, $U$ and $P$. $s_1$ is the query.
Output: Sentence score vector $f$ and topic score vector $g$.
BEGIN
1. Initialization, k=0:
        $f^0 = (1, 0, 0, ..., 0)^T$, $g^0 = (0, 0, ..., 0)^T$
2. Update sentence score vector
        $f^{k+1} = aUf^k + \frac{1}{2}(1-a)(\hat{W} + \bar{W}^T)g^k$
3. Update topic score vector
        $g^{k+1} = aPg^k + \frac{1}{2}(1-a)(\hat{W}^T + \bar{W})f^k$
4. fix the score of query in $f^{k+1}$ to 1.
5. k=k+1 Go to Step 2 until convergence.
END
```

Figure 5: Sentence Ranking Algorithm

```
Input: The sentence set $S = \{s_1, s_2, ..., s_N\}$, sentence score vector $f$
Output: Summary Y.
BEGIN:
1. Initialization: $Y = \Phi$, $X = \{S - s_1\}$.
2. while word num is less than 100:
    (a) $s_m = \arg\max_{s_i \in X} f(s_i)$
    (b) If $sim(s_m, s) < Th_{sem}$, for all $s \in Y$:
            $Y = Y + \{s_m\}$
    (c) $X = x - \{s_m\}$
END
```

Figure 6: Sentence Selection Algorithm

3.1 Summary Generation

Sentence compression can largely improve summarization quality (Zajic et al, 2007; Peng et al, 2011). Since sentence compression is not the main task in this paper, we just use the revised sentence compression techniques in (Li et al, 2011).Here, we remove the redundant modifiers such as adverbials, relative clause modifiers, abbreviations, participials and infinitive modifiers for each sentence.

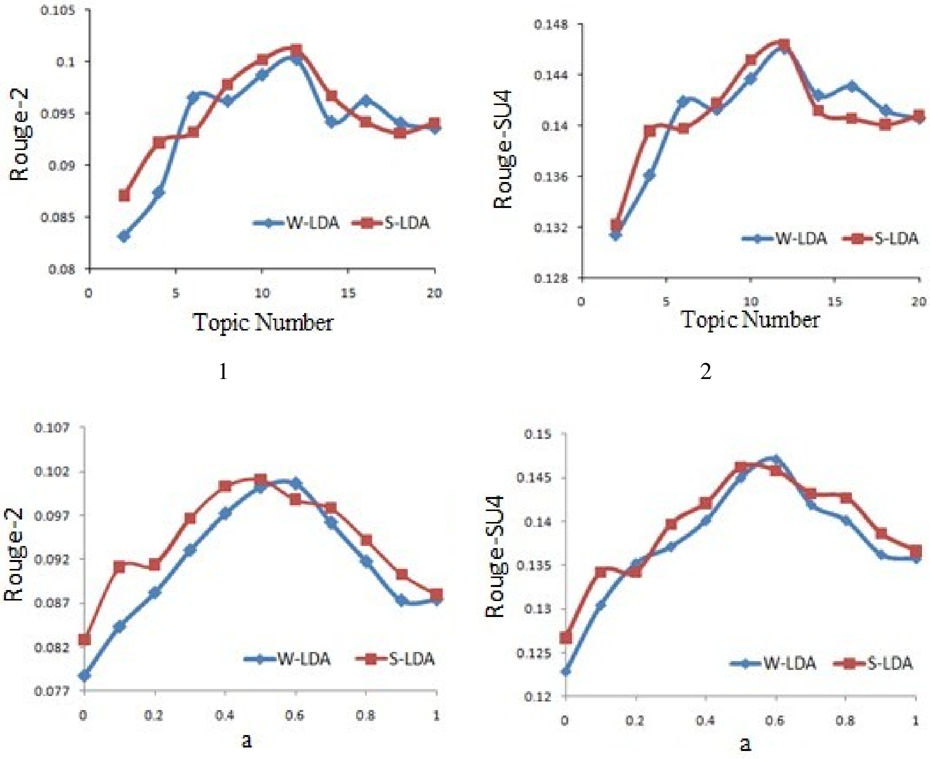
As for the sentence selection process, sentences with higher ranking score are selected into the summary. Then Maximum Marginal Relevance (MMR)(Goldstein et al, 1999) is further used for redundancy removal. We just apply a simple greedy algorithm for sentence selection as shown in Figure 6. We use Y to denote the summary set which contains the selected summary sentences. The algorithm first initializes Y to Φ and X as the set $\{S - s_1\}$. During each iteration, we select the highest ranked sentence s_j from the sentence set X. We need to assure that the value of semantic similarity between two sentences is less than Th_{sem}. Th_{sem} denotes the threshold for the cosine similarity between two sentences and is set to 0.5 in our model.

4 Experiments

The query-focused MDS task defined in TAC (Text Analysis Conference) evaluations requires generating a concise and well organized summary for a collection of related documents according to a given query. The query usually consists of a narrative/question sentence. Our experiment data is composed of TAC (2008-2009) data[2], which contain 48 and 44 document collections respectively. We use docset-A data sets in TAC which has 10 documents per collection. The average numbers of sentences per document in TAC2008 and TAC2009 are 252 and 243 respectively, and the system-generated summary is limited to 100 words. It is noted that the corpus of TAC2008 and TAC2009 are similar. In our experiment, we apply the optimal topic number trained on TAC2008 dataset to TAC2009 dataset.

[1] In our experiments, if $|f_i^k - f_1^{k+1}| \leq 0.0001(1 \leq i \leq N)$ and $|g_i^k - g_1^{k+1}| \leq 0.0001(1 \leq i \leq T)$, iteration stops.

[2] TAC data sets are for the update summarization tasks, where the summarization for docset-A can be seen the query-focused summarization task referred in this paper.



1 2

3 4

Figure 7: ROUGE score via (1)(2) topic number and (3)(4) parameter a on TAC2008.

As for evaluation metrics, we use ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin, 2004) measures. ROUGE measures summary quality by counting overlapping units such as the n-gram, word sequences and word pairs between the candidate summary and the reference summary. We report ROUGE-1, ROUGE-2, and ROUGE-SU4[3] scores and their corresponding 95% confidential intervals, to evaluate the performance of the system-generated summaries. As a preprocessing step, stopwords are firstly removed with a list of 598 stop words and the remaining words are then stemmed using PorterStemmer.[4].

4.1 Parameter Tuning

There are two parameters to tune in our model. The first parameter is a in Equation (11) that controls the tradeoff between influence from topics and from sentences. The second one is the topic number K in LDA topic model. The combination of the two factors makes it hard to find a global optimized solution. So we apply a gradient search strategy. At first, parameter a is fixed to a given value. Then the performance of using different topic numbers is evaluated. After that, we fix the topic number to the value which has achieved the best performance, and conduct experiments to find an appropriate value for a. Here, we use TAC2008 as training data and test our model on TAC2009.

First, a is set to 0.5, then we change topic number K from 2 to 20 at the interval of 2. The ROUGE score reaches their peaks when the topic number is around 12, as shown in Figure 7(1) and Figure 7(2). Then we fix the number of K to 12 and change the value of parameter a from 0 to 1 with the interval of 0.1. When the value of a is set to 0, the model degenerates into a one-layer graph ranking algorithm where topic clustering information is neglected. As we can see from Figure 7(3) and Figure 7(4) , the ROUGE scores reach their peaks around 0.6 and then drop afterwards. Thus, the topic number is set to 12 and a is set to 0.6 in the test dataset.

[3] Jackknife scoring for ROUGE is used in order to compare with the human summaries.

[4] http://tartarus.org/martin/PorterStemmer/

4.2 Baseline Comparison

We firstly compare W-LDA and S-LDA with other clustering approaches. To be fair, we use the identical sentence compression techniques and preprocessing methods for all baselines. Summaries are truncated to the same length of 100 words.

Standard-LDA: A simplified version of W-LDA without considering the background or document-specific information.

K-means: Using the K-means clustering algorithm for graph construction. We firstly randomly select K sentences as initial centroid for clusters and then iteratively assign a sentence to each cluster. The centroid is recomputed until convergence. The similarity between nodes in the graph (sentence or cluster) is computed using the standard cosine measure based on the tf-idf information. K is set to 12, the same as topic number in LDA.

Agglomerative: a bottom-up hierarchical clustering algorithm and starts with the sentences as individual clusters and, at each step, merges the most similar or closest pair of clusters, until the number of the clusters reduces to the desired number $K = 12$.

Divisive: a top-down hierarchical clustering algorithm and starts with one, all-inclusive cluster and, at each step, splits the largest cluster until the number of clusters increases to the desired number K, $K = 12$.

Approach	Rouge-1	Rouge-2	Rouge-SU4
W-LDA	0.3791 (0.3702-0.3880)	0.1092 (0.1047-0.1135)	0.1382 (0.1350-0.1414)
S-LDA	**0.3802 (0.3721-0.3883)**	**0.1109 (0.1061-0.1157)**	**0.1398 (0.1342-0.1454)**
Standard LDA	0.3702 (0.3614-0.3790)	0.1012 (0.0960-0.1064)	0.1292 (0.1242-0.1344)
K-means	0.3658 (0.3582-0.3734)	0.1046 (0.0992-0.1080)	0.1327 (0.1263-0.1391)
Agglomerative	0.3681 (0.3612-0.3750)	0.1042 (0.091-0.1093)	0.1319 (0.1266-0.1272)
Divisive	0.3676 (0.3610-0.3742)	0.1021 (0.0981-0.1061)	0.1320 (0.1275-0.1365)

Table 1: Comparison with other clustering baselines.

Table 1 presents the performance of different clustering algorithms for summarization. Traditional clustering algorithms such as K-means, Agglomerative and Divisive clustering achieve comparative results. Compared with traditional clustering algorithms, LDA based models (W-LDA, S-LDA, Standard-LDA) achieve better results. This can be explained by the clear and rigorous probabilistic interpretation of topic models. Background information and document-specific information would influence the performance of topic modeling (Chemudugunta et al, 2006), that is why S-LDA and W-LDA achieve better ROUGE performance than the standard LDA. We can also see that S-LDA is slightly better than W-LDA in regard with ROUGE performance. The reason can be explained as follows: The aim of topic modeling in this task is to cluster sentences according to their topics. So treating sentence as a unit in topic modeling would be better than treating it as a set of independent words. In addition, forcing the words in one sentence to share the same aspect topic can ensure semantic cohesion of the mined topics.

Next, we compare our model with the following widely used summarization approaches.

Manifold: One-layer graph-based semi-supervised approach developed by Wan et al.(2008). Sentence relations are calculated according to $tf - idf$ and topic information is neglected.

LexRank: An unsupervised graph-based summarization approach(Erkan and Radev, 2004), which is a revised version of the famous web ranking algorithm PageRank.

KL-Divergence: The approach developed by (Lin et al, 2006) by using a KL-divergence based sentence selection strategy.

$$KL(P_s||Q_d) = \sum_w P(w) log \frac{P(w)}{Q(w)} \tag{12}$$

where P_s is the unigram distribution of candidate summary and Q_d denotes the unigram distribution of document collection. Since this approach is designed for general summarization, query influence is not considered.

Hiersum: A LDA based approach proposed by (Haghighi and Vanderwende, 2009), where unigram distribution is calculated from LDA topic model in Equation (12).

MEAD: A centroid based summary algorithm by (Radev et al, 2004). Cluster centroids in MEAD consists of words which are central not only to one article in a cluster, but to all the articles. Similarity is measured by using $tf - idf$.

Approach	Rouge-1	Rouge-2	Rouge-SU4
W-LDA	0.3891 (0.3802-0.3980)	0.1192 (0.1147-0.1235)	0.1482 (0.1450-0.1514)
S-LDA	**0.3902 (0.3821-0.3983)**	**0.1209 (0.1161-0.1257)**	**0.1498 (0.1442-0.1554)**
Manifold	0.3581 (0.3508-0.3656)	0.1007 (0.0952-0.1062)	0.1267 (0.1214-0.1320)
LexRank	0.3442 (0.3381-0.3502)	0.0817 (0.0782-0.0852)	0.1106 (0.1064-0.1148)
KL-divergence	0.3468 (0.3410-0.3526)	0.0820 (0.0782-0.0858)	0.1117 (0.1073-0.1161)
Hiersum	0.3599 (0.3526-0.3672)	0.1004 (0.0956-0.1052)	0.1280 (0.1221-0.1339)
MEAD	0.3451 (0.3390-0.3512)	0.0862 (0.0817-0.0907)	0.1131 (0.1080-0.1182)

Table 2: Performance comparison with baselines

Performance is presented at Table 2. We can find that ROUGE performance of one-layer graph ranking algorithms such as Manifold and LexRank, where topic information is neglected, achieve worse results than all two-layer models where topic information is considered (See Table 1). This verifies our previous claim (Hardy et at., 2002; Harabagiu and Lacatusu, 2005; Wan and Yang, 2008) that the consideration of topic information will improve summarization performance. S-LDA and W-LDA achieve better performance than KL-divergence and Hiersum. This is because the sentence selection strategy for KL-divergence and Hiersum tries to select sentence best representing the document as shown in Equation (12), but do not consider the influence of query.

4.3 Manual Evaluation

W-LDA and S-LDA get comparative ROUGE scores. To obtain a more accurate measure to decide which approach is better, we perform a simple user study concerning the following aspects on 40 randomly selected topics in TAC2009: (1) Overall quality. (2) Focus: Whether the summary contains less irrelevant content? (3) Responsiveness: Whether the summary is responsive to the query. (4) Non-Redundancy: Whether the summary is non-redundant. Each respect is rated from 1 (very poor) to 5 (very good). Four native speakers who are Ph.D. students in computer science (none are authors) performed the task.

The average score and standard deviation for W-LDA and S-LDA are displayed in Table 3. We can see that the two models almost tie in foucs and non-redundancy. This is because two models use the same sentence selection strategy based on MMR for redundancy removal and propagation model to impose the query's influence on sentences. S-LDA outperforms W-LDA in overall ranking and responsiveness ranking. This implies that treating sentence as a unit in topic modeling would be preferable to just treating it as a series of independent words.

	S-LDA	W-LDA
Overall	3.98 ± 0.52	3.58 ± 0.55
Focus	3.65 ± 0.54	3.35 ± 0.61
Responsiveness	3.73 ± 0.43	3.38 ± 0.46
Non-Redundancy	3.48 ± 0.51	3.45 ± 0.48

Table 3: Manual evaluation for S-LDA and W-LDA.

5 Related Work

Graph-based ranking approaches have been hot these days for both generic and query-focused summarization (Zhou et al, 2003; Zhou et al, 2004; Erkan and Radev, 2004; Wan et al, 2007; Wei et al, 2008). Commonly used graph-based ranking algorithms are mainly inspired by the link analysis algorithm in web research such as PageRank (Page et al, 1999). (Wan et al, 2007) proposed the approach that treated the task of query-focused MDS as a semi-supervised learning task, in which the query is treated as a labeled node, and sentences as unlabeled nodes. Then the scores of sentences are determined from the manifold learning algorithm proposed by (Zhou et al, 2003) or the harmonic approach proposed by (Zhu et al, 2003).

It is worthy of noting that researchers have found that by considering topic level information, the summarization performance can be effectively improved (Hardy et al, 2002; Wan and Yang, 2008; Harabagiu and Lacatusu, 2005). For example, (Otterbacher et al, 2005) models documents as a stochastic graph and calculates sentence ranking scores with a topic-sensitive version of PageRank. (Wan and Yang, 2008) developed a two-layer graph by clustering sentences by using standard clustering algorithms such as K-means or agglomerate clustering. However, his algorithm is for general summarization where the influence of query is not considered.

A significant portion of recent work incorporates LDA topic models (Blei et al, 2008) in summarization tasks for their clear and rigorous probabilistic topic interpretations (Daume and Marcu, 2006; Titov and McDonald, 2008; Haghighi and Vanderwende, 2009; Mason and Charniak, 2011; Li et al, 2013a; Li et al, 2013b). (Haghighi and Vanderwende, 2009) introduced a LDA based model called Hiersum to find the subtopics or aspects by combining KL-divergence criterion for selecting relevant sentences. AYESSUM (Daume and Marcu, 2006) and the Special Words and Background model (Chemudugunta et al, 2006) are very similar to Hiersum. In the same way, (Delort and Alfonseca, 2012) tried to use LDA to model different levels of information for novelty detection in update summarization. Furthermore, (Paul and Dredze, 2013) extends their f-LDA to jointly model combinations of drug, aspect and route of administration as an exploratory tool for extractive summarization.

6 Conclusions and Future Work

In this paper, we propose a two-layer graph-based semi-supervised algorithm for query-focused MDS. Topic modeling techniques are used for sentence clustering and further graph construction. By considering different kinds of information such as background or document-specific information, our two LDA topic model extensions achieve better results than traditional clustering algorithms.

One primary disadvantage of our models is that it is hard to decide the topic number K in LDA models and how to define topic number is still a open problem in LDA topic models. From Figure 7, we can see that summarization performance is sensitive to topic number. We train the value of topic number on TAC2008 dataset and test the model on TAC2009. Such process makes sense because the corpus sizes and contents of two datasets are similar. But it would be hard to extend optimal topic number in TAC2008 to other datasets. Using non-parametric topic modeling techniques where topic number does not have to be predefined is one of our future works.

Acknowledgements

Thanks Jiwei Li for the insightful reviews and careful polishment. We also thank the three anonymous reviewers for their helpful comments. This work was partially supported by National High Technology Research and Development Program of China (No. 2012AA011101), National Key Basic Research Program of China (No. 2014CB340504), National Natural Science Foundation of China (No. 61273278), and National Key Technology R&D Program (No: 2011BAH10B04-03).

References

Edoardo M. Airoldi, Blei D M, Fienberg S E, et al. Mixed membership stochastic blockmodels[J]. In *The Journal of Machine Learning Research*, 2008, 9(1981-2014): 3.

David Blei, Andrew Ng and Micheal Jordan. 2003. Latent dirichlet allocation. In *The Journal of Machine Learning Research*.

Chaltanya Chemudugunta, Padhraic Smyth and Mark Steyers. Modeling General and Specific Aspects of Documents with a Probabilistic Topic Model.. In *Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference.*

Hal Daume and Daniel Marcu H. 2006. Bayesian Query-Focused Summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 305-312.

Jean-Yves Delort and Enrique Alfonseca. DualSum: a topic-model based approach for update summarization. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics.*

Gune Erkan and Dragomir Radev. 2004. Lexrank: graph-based lexical centrality as salience in text summarization. In *Journal of Artificial Intelligence Research.*

Jade Goldstein, Mark Kantrowitz, Vibhu Mittal and Jaime Carbonell. 1999. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval.*

Amit Gruber, Yair Weiss and Michal Rosen-Zvi. Hidden topic Markov models. In *International Conference on Artificial Intelligence and Statistics.* 2007

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362370.

Sanda Harabagiu and Finley Lacatusu. 2005. Topic themes for multi-document summarization. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval.*

Hilda Hardy, Nobuyuki Shimizu, Tomek Strzakowski, Liu Ting, Xinyang Zhang and Bowden Wize. 2002. Cross-document summarization by concept classification. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval.*

Feng Jin, Minlie Huang, and Xiaoyan Zhu. 2010. The summarization systems at tac 2010. In *Proceedings of the third Text Analysis Conference*, TAC-2010.

Jiwei Li and Sujian Li. 2013. Evolutionary Hierarchical Dirichlet Process for Timeline Summarization. In *ACL* 2013.

Jiwei Li and Claire Cardie. 2014. Timeline Generation: Tracking individuals on Twitter. In *WWW* 2014.

Peng Li, Yinglin Wang, Wei Gao and Jiang Jing. 2011. Generating aspect-oriented multi-document summarization with event-aspect model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.*

Chin-Yew Lin, Guihong Cao, Jianfeng Gao, and Jian-Yun Nie. 2006. An information-theoretic approach to automatic evaluation of summaries. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics.*

Chin-Yew Lin. Improving summarization performance by sentence compression: a pilot study. In *Proceedings the sixth international workshop on Information retrieval with Asian languages*.

Rebecca Mason and Eugene Charniak. 2011. Extractive multi-document summaries should explicitly not contain document-specific content. In *proceedings of ACL HLT.*

Jahna Otterbacher, Gne Erkan, and Dragomir R. Radev. Using random walks for question-focused sentence retrieval. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2005.

Lawrence Page, Sergey Brin, Rajeev Motwani and Terry Winograd. 1999. The Pagerank Citation Ranking: Bringing Order to the Web. Technical report, Stanford Digital Libraries.

Michael J. Paul and Mark Dredze. Drug extraction from the web: Summarizing drug experiences with multi-dimensional topic models. In *Proceedings of NAACL-HLT.* 2013.

Wei-Ting Peng, Wei-Ta Chu, Chia-Han Chang, et al. Editing by viewing: automatic home video summarization by viewing behavior analysis[J]. In *Multimedia, IEEE Transactions on*, 2011, 13(3): 539-550.

Dragomir Radev, Allison T, Blair-Goldensohn S, et al. MEAD-a platform for multidocument multilingual text summarization[C]. In *Proceedings of the 4th International Conference on Language Resources and Evaluation,* 2004.

Ivan Titov and Ryan McDonald. 2008. Modeling on- line reviews with multi-grain topic models. In *International World Wide Web Conference*.

Xiaojun Wan, Jianwu Yang and Jianguo Xiao. 2007. Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of International Joint Conference on Artificial Intelligence*.

Xiaojun Wan and Jianwu Yang. 2008. Multi-document Summarization using cluster-based link analysis. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*.

Furu Wei, Wenjie Li, Qin Lu, and Yanxiang He. 2008. Query-sensitive mutual reinforcement chain and its application in query-oriented multi-document summarization. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*.

David Zajic, et al. Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. In *Information Processing & Management* 43.6 (2007): 1549-1570.

Dengzhong Zhou, Jason Weston, Arthur Gretton, Olivier Bousquet and Bernhard Schlkopf. 2003. Ranking on Data Manifolds. In *Proceedings of the Conference on Advances in Neural Information Processing Systems*.

Dengyou Zhou, Olivier Bousquet, Thomas Navin and JasonWeston. 2004. Learning with Local and Global Consistency. In *Proceedings of Advances in neural information processing systems*.

Xiaojin Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and functions. In *Proceedings of the 20th International Joint Conference on Machine Learning*, 2003.

APPENDIX

To optimize $\Omega(f,g)$, shown in Equation (10), we set the partial derivative with respect to f_m to 0, for $m \in [1,N]$. Let δ_{mn} denote the index function as follows:

$$\delta_{mn} = \begin{cases} 1 & if\ m = n \\ 0 & if\ m \neq n \end{cases}$$

$$\begin{aligned}
0 &= \frac{\partial \Omega(f,g)}{f_t} \\
&= 2a\sum_{i,j} U_{i,j}(f_i - f_j)(\delta_{it} - \delta_{jt}) + 2(1-a) \\
&\times \sum_{i,j} \hat{W}_{ij}(f_i - g_j)\delta_{it} - 2(1-a)\sum_{i,j} \bar{W}_{ij}(g_i - f_j)\delta_{jt} \\
&= 2(1-a)\sum_{j} \hat{W}_{tj}(f_t - g_j) + 2(1-a)\sum_{i} \bar{W}_{it}(g_i - f_t) \\
&+ 2a\sum_{j} U_{tj}(f_t - f_j) + 2a\sum_{i} U_{it}(f_i - f_t) \\
&= f_t[4a\sum_{j} U_{tj} + 2(1-a)\sum_{j} \hat{W}_{tj} + 2(1-a)\sum_{j} \bar{W}_{jt}] \\
&- 4a\sum_{j} U_{tj}f_j - 2(1-a)\sum_{j} \hat{W}_{tj}g_j - 2(1-a)\sum_{j} \bar{W}_{jt}g_j \\
&\sum_{j} U_{tj} = 1 \qquad \sum_{j} \hat{W}_{tj} = 1 \qquad \sum_{j} \bar{W}_{jt} = 1 \\
f_t &= a\sum_{j} U_{tj}f_j + \frac{1}{2}(1-a)[\sum_{j}(\hat{W}_{tj} + \bar{W}_{jt})g_j]
\end{aligned}$$

So we have:

$$f = aUf + \frac{1}{2}(1-a)(\hat{W} + \bar{W}^T)g$$

A similar approach is used to obtain the second part of Equation (11).